

This is a postprint version of the following published document:

K. Kondepu, A. Giorgetti, F. Giannone, A. Marotta, F. Cugini, P. Castoldi, and L. Valcarenghi. (november, 2017). *SDN-enabled Latency-Guaranteed Dual Connectivity in 5G RAN*. Paper submitted in Asia Communications and Photonics Conference, Guangzhou.

DOI: [10.1364/ACPC.2017.M3B.4](https://doi.org/10.1364/ACPC.2017.M3B.4)

© 2017 Optical Society of America. One print or electronic copy may be made for personal use only. Systematic reproduction and distribution, duplication of any material in this paper for a fee or for commercial purposes, or modifications of the content of this paper are prohibited.

SDN-enabled Latency-Guaranteed Dual Connectivity in 5G RAN

K. Kondepu¹, A. Giorgetti¹, F. Giannone¹, A. Marotta²,
F. Cugini³, P. Castoldi¹, L. Valcarenghi¹

¹Scuola Superiore Sant'Anna, Pisa, Italy; ²Università degli Studi dell'Aquila, L'Aquila, Italy; ³CNIT, Pisa, Italy

Abstract: A novel SDN-controlled E-UTRAN interacting with 5G Radio Resource Management (RRM) featuring Dual Connectivity (DC) is proposed and experimentally demonstrated. Radio bearers are dynamically steered to different evolved NodeBs, to guarantee effective per-flow latency performance.

1. Introduction

Dual Connectivity (DC) is included in 3GPP technical specification of Evolved Universal Terrestrial Radio Access Network (E-UTRAN) [1] with the aim of improving the User Equipment (UE) throughput [2]. In DC, a UE is allowed to utilize radio resources provided by two distinct schedulers located in two evolved NodeBs (eNBs) (i.e., a Master eNB – MeNB — and a Secondary eNB – SeNB) connected via a non-ideal backhaul over the X2 interface.

In DC, the control plane (C-plane) communication is carried by bearers set up between Mobile Management Entity (MME) and the MeNB (i.e., S1-C interface signalling) and between the MeNB and the SeNB (i.e., X2-C interface signalling). On the other hand, data plane communication can be performed in three different ways. In *Master Cell Group (MCG) bearers*, the S1-U connection for the corresponding bearer(s) to the Serving Gateway (S-GW) is terminated in the MeNB and the SeNB is not involved in transmitting data to the UE. In *split bearers*, the S1-U connection to the S-GW is terminated in the MeNB and Packet Data Convergence Protocol (PDCP) data is transferred between the MeNB and the SeNB via the X2-U interface. Thus both MeNB and SeNB are involved in transmitting data to the UE. In *Secondary Cell Group (SCG) bearers*, the SeNB is directly connected with the S-GW via S1-U. Thus the MeNB is not involved in transmitting data to the UE.

The decision to which cells a UE shall connect is taken by Radio Resource Management (RRM) functions (i.e., UE cell association – CA – and packet scheduling – PS) implemented in the eNB [3].

Usually, when DC is considered, cell association decisions are taken on power-based metrics [4], such as Reference Signal Received Power (RSRP) and Received Signal Strength Indicator (RSSI) on each component carrier. However, other Quality of Service (QoS) constraints, such as latency, are increasingly important for some safety-critical applications (e.g., emergency communications in the vehicle to X infrastructure – V2X). In addition, bearer routing is currently based on the IP protocol. However, end-to-end latency depends not only on packet scheduling but also on traffic bearer routing.

This paper proposes for the first time and experimentally demonstrates a DC implementation in which RRM DC-related functions are co-located with an Software Defined Networking (SDN)-based Radio Access Network (RAN) controller to jointly decide to which eNBs UEs shall connect and where to route user plane (i.e., S1-U) bearers. The decision is based upon real time measurement of QoS parameters (e.g., latency) provided by the eNBs. The proposed solution allows to dynamically forward different data flows with different QoS constraints to different eNBs.

2. Proposed System Model

Fig. 1(a) shows the proposed system model. The system is based on the Long Term Evolution-Advanced (LTE-A) architecture consisting of E-UTRAN and Evolved Packet Core (EPC) including the Serving Gateway (S-GW). The E-UTRAN consists of OpenFlow switches. An SDN-based RAN controller (CU) communicates with the eNBs, the E-UTRAN switches, and the EPC. RRM DC functions are assumed to be co-located with the CU.

Periodically, the CU receives per flow QoS (e.g., latency) measurements performed by the eNBs. The measurements related to Layer 2 packet delay are performed by the eNBs at the PDCP Service Access Point (SAP), as specified in [5]. Moreover the CU is assumed to be aware of the delay experienced by the packets carried by the S1-U bearers between S-GW and eNB. The CU is also aware of the QoS constraints of the flow destined to/generated by a UE. By comparing the available measurements and the specific UE constraints the CU forwards the flow to the eNB(s) that is capable of

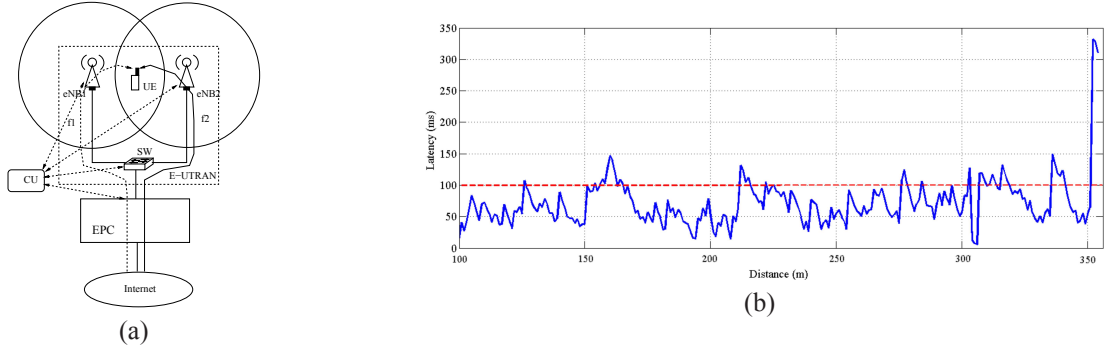


Fig. 1. (a) System model; (b) Latency vs user equipment position.

guaranteeing them. For example, if latency requirements are not met when the UE is attached to eNB1, the flow S1-U is redirected from eNB1 (i.e., S1-Ua) to eNB2 (i.e., S1-Ub) as shown in Fig. 1(a).

3. Performance evaluation scenario and results

The testbed depicted in Fig. 2 is utilized to evaluate the proposed system. The testbed includes four Linux PCs equipped with five Gigabit Ethernet interfaces. One interface is used to guarantee the connection toward the CU. The other four interfaces are included in a virtual Ethernet switch supporting OpenFlow 1.3. Each switch is implemented by using OpenvSwitch (OvS) 2.4 [7] (OvS). OvS 2 interconnects the E-UTRAN to the EPC, OvS 3 and OvS 4 are the switches to which two eNBs (i.e., eNB1 and eNB2) are connected respectively. The two eNBs, are assumed to be placed at 350 meters of distance and the UE is assumed to move at 1 meter per second between eNB1 and eNB2. OvS 1 emulates the possibility for the UE of connecting to eNB1 or eNB2. The CU is implemented on a separate Linux server using Ryu version 3.20.2 [8].

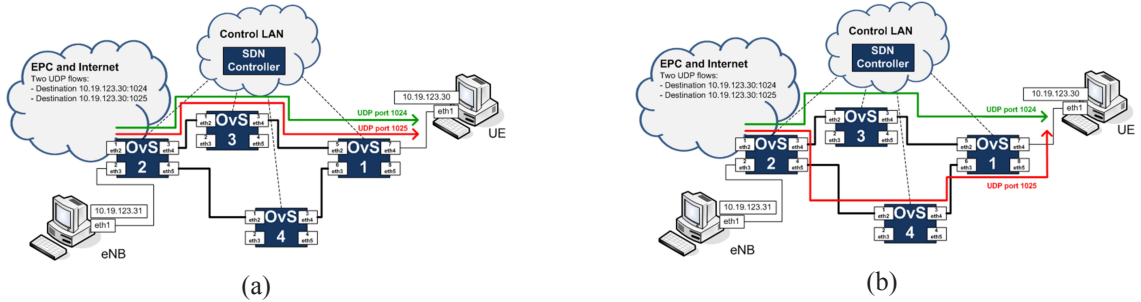


Fig. 2. SDN testbed, (a) both flow established through OvS3; (b) one flow switched through OvS4.

Two data flow arriving from the Internet are assumed to reach the UE, as depicted in Fig. 2(a). Each flow is carried by a different S1-U bearer due to the different latency constraints. Specifically in the considered scenario, two UDP flow are established toward the UE. The UDP flow are generated with a Spirent SPTN4U traffic analyzer [9]. The first flow uses UDP port 1024 and has not latency requirements, the second flow is directed to UDP port 1025 and it requires a one way latency lower than 80 ms.

To model the latency experienced by the flows when the UE moves, a simulation is performed using the open source software OpenAirInterface (OAI) [6] and considering Rel10 Access Stratum of LTE standard. In the simulation the UE is forced to be connected to only one eNB (e.g., eNB1) and, for simplicity, only one flow is considered. The obtained results are illustrated in Fig. 1(b) depicting the latency (i.e., one way delay between eNB1 and UE) experienced by the flow as a function of the distance between eNB1 and UE. The figure shows that the latency oscillates between 20 ms and 150 ms for distances less than 350 meters, then the latency starts to increase up to the value of 300 ms.

To emulate the real time latency measurements a monitoring PC that elaborates the latency data obtained from the simulation and feed them to the CU is utilized. As depicted in Fig. 2(a) the monitoring PC is connected to OvS 2 and it sends an IP packet including the latency value, averaged on the previous interval, every five seconds. In turn, OvS 2 encapsulates the received IP packet in an OpenFlow PacketIn message and forwards it to the CU.

Upon reception of the latency measure, the CU compares the received value with the specific threshold stored for each traffic flow established in the network. If a flow requires a lower latency than the experienced one, such flow is switched toward an eNB featuring a better latency.

Fig. 2(a) illustrates the initial flow paths when, at switch OvS 2, both flow are using output port 3 (i.e., interface eth4). Then, when a latency value higher than 80 ms is received, the UDP flow with port 1025 is redirected toward output port 4 (i.e., interface eth5), as depicted in Fig. 2(b). The CU also implements an hysteresis loop, to avoid to switch paths too frequently. This way the traffic flow is redirected back on the path in Fig. 2(a), when a latency less than 60 ms is measured.

No.	Time	Source	Destination	Protocol	Length	Info
471	28.421255	10.30.2.36	193.205.83.101	OpenFlow	236	Type: OFPT_PACKET_IN
472	28.422296	193.205.83.101	10.30.2.36	OpenFlow	106	Type: OFPT_PACKET_OUT
505	30.649369	10.30.2.36	193.205.83.101	OpenFlow	236	Type: OFPT_PACKET_IN
506	30.651412	193.205.83.101	10.30.2.36	OpenFlow	178	Type: OFPT_FLOW_MOD
508	30.651436	193.205.83.101	10.30.2.36	OpenFlow	106	Type: OFPT_PACKET_OUT
547	33.977331	10.30.2.36	193.205.83.101	OpenFlow	190	Type: OFPT_PACKET_IN
548	33.977467	10.30.2.36	193.205.83.101	OpenFlow	190	Type: OFPT_PACKET_IN
550	33.979223	193.205.83.101	10.30.2.36	OpenFlow	106	Type: OFPT_PACKET_OUT
552	33.979248	193.205.83.101	10.30.2.36	OpenFlow	106	Type: OFPT_PACKET_OUT
560	34.389677	10.30.2.36	193.205.83.101	OpenFlow	190	Type: OFPT_PACKET_IN
561	34.390884	193.205.83.101	10.30.2.36	OpenFlow	106	Type: OFPT_PACKET_OUT
613	38.416228	10.30.2.36	193.205.83.101	OpenFlow	236	Type: OFPT_PACKET_IN
614	38.418029	193.205.83.101	10.30.2.36	OpenFlow	178	Type: OFPT_FLOW_MOD
616	38.418054	193.205.83.101	10.30.2.36	OpenFlow	106	Type: OFPT_PACKET_OUT
672	42.135472	10.30.2.36	193.205.83.101	OpenFlow	190	Type: OFPT_PACKET_IN

<pre>0 = Has mask: False Length: 4 Value: 10.10.122.30 OUT Field Class: OFPT_PACKET_IN_BASIC (0x0000) 0010 0000 = Field: OFPT_PACKET_IN_UDP_PORT (0)0 = Has mask: False Length: 2 Value: 1025 </pre>	<pre>0 = Has mask: False Length: 4 Value: 10.10.122.30 OUT Field Class: OFPT_PACKET_IN_BASIC (0x0000) 0010 0000 = Field: OFPT_PACKET_IN_UDP_PORT (0)0 = Has mask: False Length: 2 Value: 1025 </pre>	<pre>0 = Has mask: False Length: 4 Value: 10.10.122.30 OUT Field Class: OFPT_PACKET_IN_BASIC (0x0000) 0010 0000 = Field: OFPT_PACKET_IN_UDP_PORT (0)0 = Has mask: False Length: 2 Value: 1025 </pre>
--	--	--

Fig. 3. Capture of OpenFlow messages exchanged between OvS2 and CU.

Fig. 3 illustrates the capture of the OpenFlow messages exchanged between OvS 2, whose control interface IP address is 10.30.2.36, and the CU, whose IP address is 193.205.83.101, to implement flow switching. The time is measured in seconds. As shown in the figure, at time 30.649369 s the OFPT_PACKET_IN message is sent by the OpenFlow switch to the CU. For evaluation purpose only, IPv4 protocol time to live (TTL) field is used to send the latency values obtained with OAI simulations. The latency value is set to 90 ms in the first OFPT_PACKET_IN message, which is higher than the flow latency requirement (i.e., 80 ms). Thus, the SDN controller modifies the OvS 2 flow entries by using an OFPT_FLOW_MOD message. More specifically, as detailed in the leftmost inset, the flow matching fields are the destination IP address and UDP port number. The second inset shows the successful configuration of the flow entry by setting the OvS 2 output port 4 (i.e., eth5). After that, at time 38.416228 s, a OFPT_PACKET_IN message, carrying a latency value of 50 ms, which is lower than the the flow latency requirement, is sent to the SDN controller. Hence, the second OFPT_FLOW_MOD message switches the flow back. The third inset shows the successful flow configuration by setting the OvS 2 output port 3 (i.e., eth4). The time elapsing between when the OvS 2 sends the OFPT_PACKET_IN message and when the SDN controller sends the OFPT_FLOW_MOD message is about 2 ms. Moreover, during both reconfigurations of the switch OvS 2 no packets are lost.

4. Conclusions

This paper proposed the integration of an SDN controller and dual connectivity functions in a 5G radio access network for guaranteeing the requested latency to flows destined to a user equipment. The experimental evaluation showed that the proposed architecture was capable of steering the flows between cells serving the user in few milliseconds while guaranteeing the requested latency and without any packet loss.

5. Acknowledgements

This work was partly funded by the project H2020-ICT-2014-1 “Wishful” (645274), and partly by EU H2020 5G-TRANSFORMER project (761536).

References

1. “3GPP; E-UTRA and E-UTRAN; Overall description; Stage 2 (Release 14)”, 3GPP TS 36.300, v14.0.0 (2016-09)
2. NGMN, Guidelines for LTE Backhaul Traffic Estimation, NGMN Alliance (2011)
3. B. Tripathi *et al.*, “LTE E-UTRAN and its Access Side Protocols”, Radisys, White paper, 2011
4. Hua Wang *et al.*, “Dual connectivity for LTE-advanced heterogeneous networks”, Wireless Networks, vol. 22, no. 4, 2016
5. “3rd Generation Partnership Project; E-UTRA; Layer 2 - Measurement (Release 13)”, 3GPP TS 36.314, v13.1.0
6. N. Nikaein *et al.*, “OpenAirInterface: a flexible platform for 5G research”, Proc. SIGCOMM (2014).
7. OpenvSwitch, www.openvswitch.org.
8. Ryu SDN controller, [github.io/ryu/](https://github.com/ryu/ryu).
9. Spirent, <http://www.spirent.com/>